A Multilevel Classification Method for Classifying Web Pages

Bharadwaj J(CS13M058) and
Pankaj Kashyap(CS13M035)

Dept. of CSE, IIT Madras

Abstract. Given, a large set of web pages in the retail domain, our aim is to classify them into Product Pages, Product Listing Pages and Irrelevant Pages. We had earlier proposed a multilevel classification scheme for the same, where we attempt to classify based on the less informative parts of the web page, and if unsuccessful, start looking at the page content. The scope of this report, is restricted to the latter, where we classify based on the web page content only. Also, we focus on the elimination of irrelevant pages, by constructing one-class Support Vector Machine classifiers on the Product page and Listing page data. This can be viewed as a *Novelty Detection Problem*, where the irrelevant pages do not have any model for themselves, but are treated as outliers or abnormalities.

1 Introduction

One Class Support Vector Machines was first proposed by Schölkopf et al [1]. This method constructs a function, that is positive for a small subset of the data and negative everywhere else. This can be visualized, as a small ball or hypersphere around the positive class, separating them from the negative class. This is particularly useful when the negative class is extremely large, compared to the positive data or when it is not possible to generate sufficient data for one of the two classes to train a binary classifier. In our problem, the irrelevant pages are the negative pages. They include all web pages that are not product or product listing pages, which is massive compared to the positive class. Training a model, that captures features of all such pages is almost impossible. Hence, they are treated as outliers and the problem itself is reduced to a Novelty Detection Problem.

The data used for training consisted of 2875 web pages, out of which 1433 were product pages and the remaining 1442 were product listing pages. These were obtained from around 300 different retailers. Also for testing, 50 different irrelevant pages were obtained from popular retail websites like flipkart, amazon and ebay. These included pages such as user information pages, career pages, shipping rates, policies, press releases etc. Natural Language Toolkit was used for extracting the features and Scikit Learn package, based on LIBSVM [3] was used for training the one-class SVM on the product and product listing pages.

2 Methodology

The entire web page classification process can be divided into the following four stages: *Document-Vector matrix creation, feature extraction, training classifiers on the reduced data, testing on new data.*

2.1 Document-Vector model

Document-Vector model is used to represent the presence or frequency of words in a set of documents. It is called bag-of-words representation, where the rows of a matrix represent a document, and the columns represent the words in the document. The cell values can be either 0/1 indicating absence or presence of words in documents or they can also hold the frequency of words in documents. In this experiment, we use the 0/1 based bag of words model, where each row represents a unique web page and the columns represent the words in the web pages. Thus, we generate two sets of document-vector matrices: one for the product pages and one for product-listing pages.

2.2 Feature Extraction

For a html document features can be of two types: *Text Features,* captured by the bag of words model, and *Structural Features.* The structural features themselves, might not carry much information about the web page. However, when combined with the text features,they are expected to improve the results. At present, the focus is on text features only. However, in future, ways to combine both these sets of features and train the model on the combined feature set will be explored.

The html document may contain stop words, punctuation marks, digits and empty spaces, which do not contribute to the classification. Hence, it is necessary to remove features corresponding to these, while vectorizing the document.

2.3 Training and Testing

The first matrix will have 1483 rows, corresponding to 1433 product pages and 50 test pages and the second matrix will contain 1492 rows, corresponding to 1442 listing pages and the same 50 web pages. For training the product pages model, we took out 133 product pages and used them as test data along with the generated test pages. Similarly, for the listing pages 150 were taken out for testing. It is necessary to shuffle the pages, before taking out the test pages, so as to have a mix of all retailers in the training data.

One class SVMs using Radial Basis Function kernel [2], were trained on these two datasets. Model parameters ν and γ were varied and the models generated were tried on the test data taken out. Since, we need to minimize both false positives and false negatives, the model with the highest f-measure was considered as the best model and its results are shown below.

3 Experiments and Results

The results indicate that, the classifier developed with the listing pages as the positive class performs better that with the product pages as the positive class.

Table 1. Predictions on the test data

Positive class	Total Positives	Total Negatives	TP	TN	FP	FN
Product Page	133	50	115	32	18	18
Listing Page	150	50	131	41	1	19

Table 2. Precision, Recall and F1-Score measures on the test data

Positive class	Precision	Recall	f-Measure
Product Page	0.865	0.865	0.865
Listing Page	0.873	0.992	0.929

4 Conclusions and Future Work

The classifiers developed are adequate for eliminating irrelevant web pages. However, the biggest challenge is develop a classifier that will differentiate between product web pages and product-listing web pages. There is a large overlap between the two classes, as the product listing pages themselves may contain descriptions of a number of products. Hence, content specific features alone are not sufficient to differentiate between the two classes. We need to extract features that carry information about the structure of the web pages. One such approach would be, to use the html tag distribution by calculating the tag to text ratio in web pages.

In addition, the multilevel scheme of classification also needs to be implemented. Along with a series of classifiers at each level, we also need to develop a confidence measure that indicates the efficiency of classification at each level. On close examination of the web page URLs, one can be assured that this will produce good results for a good percentage of product and listing pages, as the URLs themselves carry text with sufficient discriminating power. For example, many of the product page URLs will have the text 'productId=' in the URL. Similarly, many product listing pages have 'categoryId' in their URLs. The confidence measure should be sufficiently high for such classifications.

References

[1] B. Schölkopf, R.C. Williamson, A.J. Smola, J.Shawe-Taylor and J.C. Platt. Support Vector Method for Novelty Detection . Technical report, Microsoft Research, MSR-TR-99-87, 1999.

[2] L. M. Manevitz and M. Yousef. One-class SVMs for document classification. Journal of Machine Learning Research,2:139-154, 2001

[3] C.C. Chang and C. J. Lin. LIBSVM: A Library for Support Vector Machines (2013)